AUTOBLOCKER: A SYSTEM FOR DETECTING AND BLOCKING OF NETWORK SCANNING BASED ON ANALYSIS OF NETFLOW DATA.

A.Bobyshev, D.Lamore, P.Demar, FNAL, Batavia, IL 60510, USA

Abstract

In a large campus network, such at Fermilab, with tens of thousands of nodes, scanning initiated from either outside of or within the campus network raises security concerns. This scanning may have very serious impact on network performance, and even disrupt normal operation of many services. In this paper we introduce a system for detecting and automatic blocking excessive traffic of different kinds of scanning, DoS attacks, virus infected computers. The system, called AutoBlocker, is a distributed computing system based on quasi-real time analysis of network flow data collected from the border router and core switches. AutoBlocker also has an interface to accept alerts from IDS systems (e.g. BRO, SNORT) that are based on other technologies. The system has multiple configurable alert levels for the detection of anomalous behaviour and configurable trigger criteria for automated blocking of scans at the core or border routers. It has been in use at Fermilab for about 2 years, and has become a very valuable tool to curtail scan activity within the Fermilab campus network.

SYSTEM DESCRIPTION.

The Autoblocker is designed to detect and automatically block in near real-time excessive traffic typically produced by network scanners. We call traffic excessive if it is not directly related to the mission of the laboratory and consumes significantly large amounts of the network and computing resources. It is typically caused by scanning of different natures, both inbound and outbound, denial of service attacks and other. Detection of scanners is based on analysis of the flow data gathered at the border and core of the network but can be extended by detectors that use other technologies to find traffic patterns. On average, the latency between detection of anomalous conditions and actions is within a 1 – 2 minute range. The logical architecture of the system is depicted in figure 1. It consists of

- Detector modules that can run in a distributed computing system;
- Alerts Processing Selector that deals with the alerts delivered by different detectors
- State Machine that drives all other components through work cycles
- Action Processing Module that maintains logical actions independently on the physical infrastructure
- Netconfig, a module that depends on the physical infrastructure and implements actual changes to isolate excessive traffic.

In simplified form AutoBlocker's work cycle can be described as the following steps.

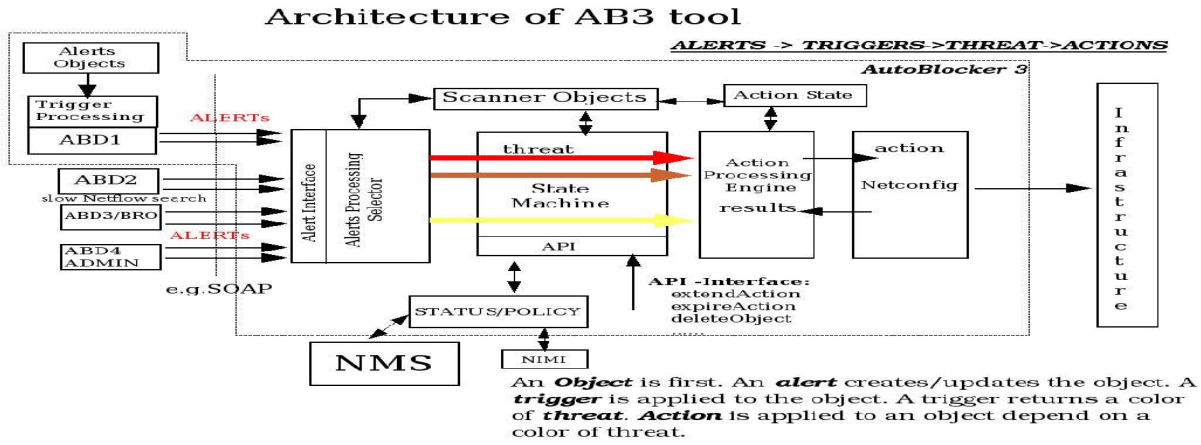

Figure 1: The logical architecture of AutoBlocker.

The anomaly detectors inspect flow data and generate alerts if events of interest occur. The key identifier of any alert is IP address of a host. For all detected hosts we calculate quantified metrics. Then, depending on the detector's type, all alerts are evaluated against the triggers that will return a color of the threat. If a color is not green then the corresponding action will be deployed.

AutoBlocker features

Major features that are currently implemented are

- multiple metrics for traffic characterization
- multiple triggers identifying unusual traffic behaviour
- multiple threats(colors) identifying different severity of occurred events
- distributed architecture, multiple anomaly detectors that can be based on different technologies not necessarily based on analysis of flow data
- multiple and expandable origins of excessive traffic, i.e offsite, onsite. Origins are treated differently
- control of multiple groups of devices with device specific configurations

A correlation between traffic patterns and scanning activity is determined by quantified metrics. For flow based detectors we use generic metrics such as a count of unique hosts contacted within a specified interval, inconsistency between inbound and outbound flows, valid DNS names of contacted hosts and many others that we identified based on the known nature of Fermilab traffic. Many of these metrics are calculated by using programs from the flow-tools package[1].

Currently the following actions are implemented and deployed automatically if triggered:

- BLOCK/UNBLOCK – maintains logical operations on blocking and unblocking of the detected scanners
- NETCONFIG – initiated in the response of BLOCK/UNBLOCK actions if it requires any configuration changes in the network infrastructure
- WATCH/resetWATCH – activates special type of triggers that deploys actions for repeatable alerts
- NOTICE – notify the data communication group and computer security team about unusual traffic conditions with detailed description and pointer to original data that allows further investigation
- NONE/flushNONE – performs so called "dry run" mode. In this mode the AutoBlocker runs similar to action BLOCK/UNBLOCK except that no actual changes in the network infrastructure are done.

Actions are based on the color of threat returned by the triggers. The map in figure 2 introduces the default

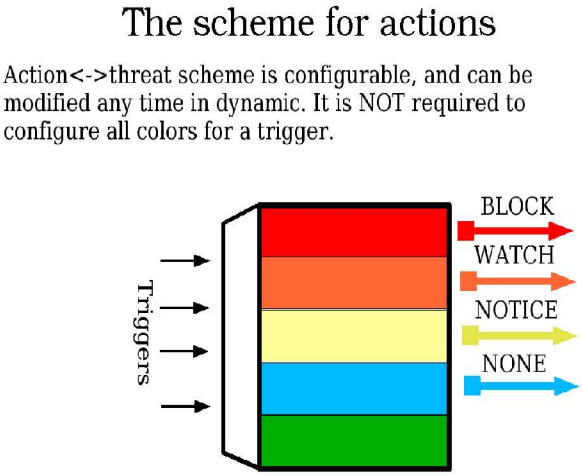

Figure 2: Example of threats to the actions map.

scheme to deploy actions. This scheme can be changed dynamically. Currently, we have twelve triggers defined, but it can be extended if necessary. There are two types of triggers, primary and secondary. The primary triggers are used to initiate actions directly according to the map in figure 2. Secondary triggers can be used in combination with other secondary or/and primary triggers to compose a primary trigger or for informative purposes. Logical expressions can be used while combining new complex triggers.

The AutoBlocker's API allows authorized clients to access the status information and control some functions via the SOAP protocol. Currently supported primitives are:

- sendAlert (sendBlockOfAlerts) interface for input alerts from distributed detectors
- expireAction - remove a system from the action
- extendAction - add the system in specified action
- objectStatus - get description of the scanner
- deleteObject - delete the object describing the scanner
- actionState – show all hosts in specified action.

THE RESULTS

The AutoBlocker has been used at Fermilab for about 2 years. It has evolved through several modifications aimed to reduce the number of false positive detections.

Often worm or virus infected machines are sources of very aggressive scanning. Fermilab has an open computing environment allowing physicists from around the world to come with personal notebooks to participate in conferences, workshops or in the experiments. Visitors from other organizations are allowed unlimited use of the network once they are on site. This makes it very complicated to prevent the propagation worms and viruses infection past the border of the network. The

AutoBlocker system shows very good performance and efficiency in such an open environment. A good demonstration is the well known spread of the SoBig, MSBlast and Welchia worms that hit many corporate networks and individual computers back in August 2003. Table 1 summarizes AutoBlocker statistics on how these worms affected the Fermilab network.

Table 1: Stats on blocks of the worm infected systems

Direction	***Blocks***	***unique hosts***	***worm infected***	***False positives***
Inbound	308196	70166	86%*	<1%*
Outbound	32082	371	350	2

- estimates are based on analysis of 1000 daily hosts

Also very important for Fermilab was the blocking of infected outbound hosts. Due to a conference with many visitors onsite we got 350 confirmed worm infections in the campus network, mostly personal notebooks. Two systems were found using the scanning technique by some of their applications to find network neighbours. The ninety systems were scanning by using ports other than NETBIOS 135-139,445 used by the mentioned above worms. A total daily traffic of outbound scanners stopped at the border was about 20-50MBytes while passed traffic was just 1-2 Mbytes. The AutoBlocker stopped 96% of traffic of Fermilab's hosts from infecting offsite machines. The computer security team was notified within 1-2 minutes about compromised computers with information helping to track and fix it.

worm infected. Approximately 13% of the blocked hosts were scanning on a mixture of ports and were within our typical background noise, and less than a 1% of all hosts were not so obvious to us. The chart in figure 3 shows traffic blocked and passed through the border from the worm infected computers.

As it should be seen in the chart a typical amount of blocked inbound traffic is a few megabytes per day. On August 11 of 2003 when SoBig worm hit Fermilab it jumped to three hundred megabytes. Because of a 1 min AutoBlocker's latency some traffic from infected systems was still able to enter the Fermilab network and potentially infect other systems. However, comparing the amount of passed and blocked traffic we may conclude that efficiency is pretty good.

REFERENCES

[1] Cisco Systems Inc. Cisco CNS NETFLOW Collection Engine http://www.cisco.com

[2] The OHIO State University , Flow-tools, http://www.splintered.net/sw/flow-tools

[3] Bro: A System for Detecting Network Intruders in Real-Time, http://www.icir.org/vern/bro-info.html

[4] SNORT: The Open Source Network Intrusion Detection System, http://www-snort.org/

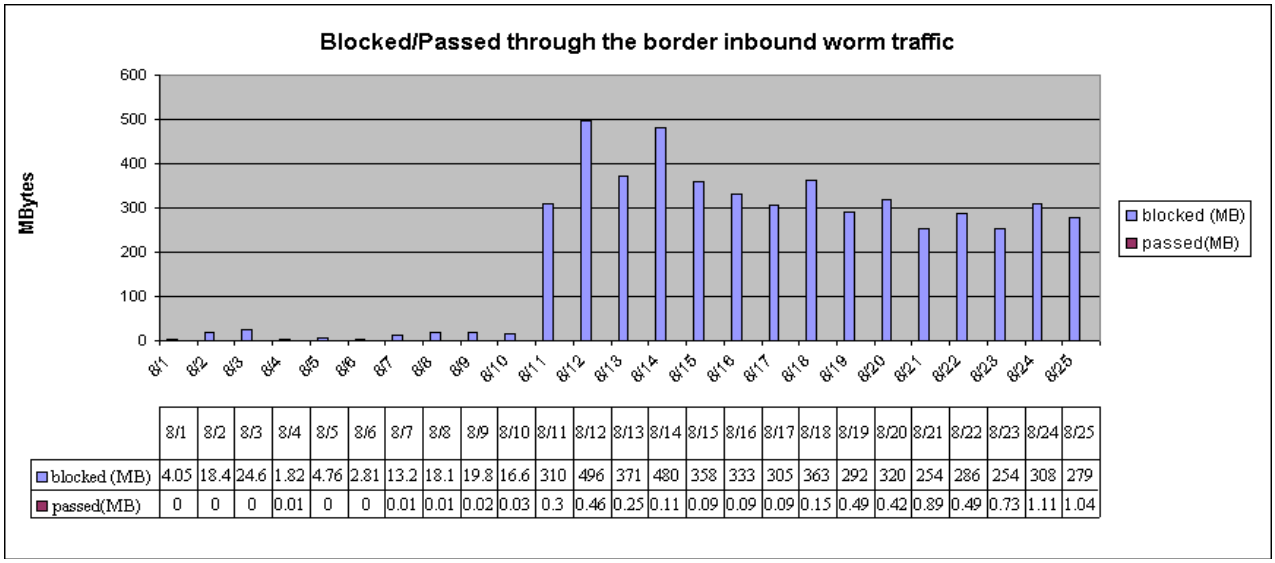

Figure 3: The AutoBlocker statistics on blocking of worm infected machines.

86% of all inbound hosts blocked by the AutoBlocker that were investigated in detail were identified as clearly